REMARKS/ARGUMENTS

Prior to this amendment, claims 1-27 were pending. In this amendment, claims 1, 17, 19 and 27 are amended, no claims are canceled and no claims are added. No new matter is added. Thus, after entry of this amendment, claims 1-27 are pending.

I. Claim Rejections – 35 USC § 103(a), Drake, Pastor, Sang-Kyun,

Claims 1, 3, 5, 6, 14, 15, 17, 19 and 25-27 are rejected under 35 U.S.C. 103(a) as being obvious over *Drake* (U.S. Publication No. 2003/0070142), in view of *Pastor* (U.S. Patent No. 6,681,383), and further in view of Sang-Kyun Kim ("*Sang")* ("Immediate and Partial Validation Mechanism for the Conflict Resolution of Update Operations in XML Databases."). Applicants respectfully submit that these references do not teach or suggest each element of these claims, as amended. This rejection is thus obviated in view of the amendments to the claims.

For example, claim 1 as amended recites in part:

> querying the database to retrieve one or more validation rules for each object in the set of objects, the one or more validation rules stored independently from the set of objects requiring validation, wherein the one or more validation rules retrieved from the database is based on the context of metadata validation

(*emphasis added*). Such a limitation is not disclosed or suggested by the cited references, alone or in combination. *Drake* does not disclose or suggest such a limitation and actually describes the opposite, wherein validation rules are encapsulated within data objects. For example, P[0018] of *Drake,* reproduced below recites that validation is coupled with or encapsulated with the data model itself.

> **[0018]** The present invention defines novel techniques for performing data validation. Validation is coupled with, or encapsulated with, the data values to which the data validation pertains, thereby becoming a part of the data model itself. This approach enables real-time data validation, as a user interacts with a data model through an executing application or GUI window interface.

As such, if the validation is coupled with the data model itself, there would be no need, nor is it suggested by *Drake*, to query a database to retrieve validation rules. A similar teaching is presented in P[0011] of *Drake* which is relied on by the Office Action, reproduced in part below, which also teaches that the validation is encapsulated with the data model.

> [0011] To achieve the foregoing objects, and in accordance with the purpose of the invention as broadly described herein, the present invention provides methods, systems, and computer program products for improving data validation. In one aspect, this technique comprises defining one or more validation criteria and encapsulating the defined validation criteria with a data model to which they apply. The technique may also comprise using the defined validation criteria to

P[0019-0020] and P[0024-0025] of *Drake* also contain language similar to that above, in which the validation rules are coupled with the data, forming a complex object. (*Drake*, P[0019]). As such, *Drake* describes the validation rules being encapsulated within the data object, and as such, there would be no need to query a database to retrieve validation rules.

Furthermore, because *Drake* describes validation rules that are encapsulated with the data model itself, it is not possible for *Drake* to disclose or suggest, "*the one or more validation rules stored independently from the set of objects requiring validation,*" because this would be the exact opposite of *Drake's* teaching. If the validation rules are encapsulated in the model, they cannot be stored independently, as the model and the validation rules are integrated as described by *Drake*.

Finally, *Drake* does not describe "*wherein the one or more validation rules retrieved from the database is based on the context of metadata validation.*" As discussed above, *Drake* describes encapsulating the validation rules with the data model. Thus, once the model is retrieved, the validation rules are necessarily present. *Drake* cannot describe retrieving rules based on the context of validation, because once the model is selected, there is no longer the

ability to retrieve different rules based on the context, as the specific rules are already necessarily present once the model is retrieved.

This discrepancy is not resolved by the addition of *Pastor*, even if there were a reason to combine *Drake* and *Pastor*. *Pastor* describes a system wherein software requirements are translated into a formal specification, and from this formal specification, software code can be generated. (*Pastor*, Abstract). *Pastor* does not describe storing rules for validation independently of the model, but rather the rules (formal specification) themselves are used to generate the model, without further intervention. (*Pastor*, Col. 3, lines 25-46). Thus, because the model is directly generated from the rules, there would be no reason to check the model against the rules, because by definition, the model is automatically generated from the rules, and is thus consistent with the rules. Additionally, even if *Pastor* described "querying the database to retrieve one or more validation rules for each object in the set of objects, "*the one or more validation rules stored independently from the set of objects requiring validation, wherein the one or more validation rules retrieved from the database is based on the context of metadata validation*" the combination with *Drake* would be improper, as it would alter the primary mode of operation of *Drake*. As described above, *Drake* stores the rules encapsulated within the model. Any modification of *Drake* to store the rules independently of the model would alter the primary mode of operation of *Drake*.

The addition of *Sang*, even if there were a reason to do so, still does not resolve this discrepancy. *Sang* describes a method for partial and complete validation of an XML document. (*Sang*, Pg. 388-390). The XML document (model) verified against the DTD (rules). (Id.). *Sang* does not describe the rules being independent of the model. In fact, *Sang* does not describe the rules being retrieved from a database at all, nor does it describe different rules being retrieved based on the context of validation. *Sang* cannot describe different validation rules based on context, because an XML must conform to the DTD at all times, otherwise it is not a valid XML document. Thus, regardless of the context of validation, the rules would necessarily remain the same. In addition, just as above, the combination of *Sang* with *Drake* would be improper as it would alter the primary mode of operation of *Drake*. As described above, *Drake* stores the rules

encapsulated within the model. Any modification of *Drake* to store the rules independently of the model would alter the primary mode of operation of *Drake*.

The addition of *Mikhailov*(U.S. Patent No. 6,968,500), even if there were a reason to do so, still does not resolve this discrepancy. *Mikhailov* teaches an automatic forms handling system (col. 1, lines 8-15; col. 5, lines 19-38), and is cited as teaching a group of types of associated metadata (Office Action, pg. 14). A combination of these references still would not result in, or provide motivation for the limitations as discussed above. In addition, just as above, the combination of *Mikhailov* with *Drake* would be improper as it would alter the primary mode of operation of *Drake*. As described above, *Drake* stores the rules encapsulated within the model. Any modification of *Drake* to store the rules independently of the model would alter the primary mode of operation of *Drake*.

The addition of *Lindberg* (U.S. Publication No. 2003/0028540) , even if there were a reason to do so, still does not resolve this discrepancy. *Lindberg* teaches a system for transferring information between a user interface and a database over a network (paragraph [0010]), and is cited as teaching a first subject as a root object for a collection of associated objects (Office Action, pg. 16). A combination of these references still would not result in, or provide motivation for the limitations as discussed above. In addition, just as above, the combination of *Lindberg* with *Drake* would be improper as it would alter the primary mode of operation of *Drake*. As described above, *Drake* stores the rules encapsulated within the model. Any modification of *Drake* to store the rules independently of the model would alter the primary mode of operation of *Drake*.

Withdrawal of the rejection of claim 1, and the claims which depend therefrom, is respectfully requested for at least the reasons set forth above. Furthermore, the remaining claims contain limitations that are likewise not taught or suggested for reasons including at least some of those set forth above with respect to claim 1. Withdrawal of the rejection of the remaining claims is respectfully requested.

II. Claim Rejections under 35 USC § 103, Drake, Rasmussen, Sang-Kyun, Mikhailov

Claims 2, 4, 18, and 20-21 are rejected under 35 U.S.C. §103(a) as being obvious over *Drake* and *Pastor* and *Sang* further in view of *Mikhailov* (US 6,968,500). These claims are not rendered obvious by *Drake* and *Pastor* and *Sang* as discussed above. Combining *Mikhailov* with these references, even if there were motivation to do so, still would not render these claims obvious. *Mikhailov* teaches an automatic forms handling system (col. 1, lines 8-15; col. 5, lines 19-38), and is cited as teaching a group of types of associated metadata. A combination of these references still would not result in, or provide motivation for the limitations as discussed above. The claims are allowable at least by virtue of their dependence from their respective independent claims. Withdrawal of this rejection is respectfully requested.

III. Claim Rejections under 35 USC § 103, Drake, Rasmussen, Sang-Kyun, Lindberg

Claims 7-13, 16, and 22-24 are rejected under 35 U.S.C. §103(a) as being obvious over *Drake* and *Pastor* and *Sang* further in view of *Lindberg* (US 2003/0028540). These claims are not rendered obvious by *Drake* and *Pastor* and *Sang* as discussed above. Combining *Lindberg* with these references, even if there were motivation to do so, still would not render these claims obvious. *Lindberg* teaches a system for transferring information between a user interface and a database over a network (paragraph [0010]), and is cited as teaching a first subject as a root object for a collection of associated objects. A combination of these references still would not result in, or provide motivation for the limitations as discussed above. The claims are allowable at least by virtue of their dependence from their respective independent claims. Withdrawal of this rejection is respectfully requested.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this Application are in condition for allowance and an action to that end is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 415-576-0200.

Respectfully submitted,

/Preetam B. Pagar/

Preetam B. Pagar
Reg. No. 57,684

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 415-576-0200
Fax: 415-576-0300
PBP:deb
62507211 v1